

Amendments to the Specification:

Please replace Paragraph 13 of the instant specification with the below marked up version. A clean version of Paragraph 13 is provided below.

Marked up version:

[0013] Accordingly, example command (a) would be entered by a user desirous of printing the contents of a file named “file1”. Once this print command is executed, the user is then prompted to enter a new command. In the case ~~[[were]]~~ where a user is desirous of printing the contents of several different files, the user is required to enter a corresponding number of print commands according to the command format introduced as command (a). Typically, user interaction dwells until a pending command is fully executed. As such, a user will need to enter these commands sequentially, waiting for a new command prompt once a previously entered command is fully executed.

Clean version:

[0013] Accordingly, example command (a) would be entered by a user desirous of printing the contents of a file named “file1”. Once this print command is executed, the user is then prompted to enter a new command. In the case where a user is desirous of printing the contents of several different files, the user is required to enter a corresponding number of print commands according to the command format introduced as command (a). Typically, user interaction dwells until a pending command is fully executed. As such, a user will need to enter these commands sequentially, waiting for a new command prompt once a previously entered command is fully executed.

Please replace Paragraph 19 of the instant specification with the below marked up version. A clean version of Paragraph 19 is provided below.

Marked up version:

[0019] Fig. 4 is a flow diagram that depicts additional alternative embodiments of a method for determining a maximum quantity of parallel processing threads. According to one such additional alternative embodiment, one or more environment variables (step 55) maintained by the operating system are utilized to establish a maximum quantity of parallel processing threads that an application embodying the present method is allowed to create. It is difficult to enumerate all of the various types of environment variables that can be maintained by any particular embodiment of an operating system. However, the scope of the claims appended hereto is intended to encompass derivative methods wherein the maximum quantity of parallel processing threads that can be created by an application is established according to one or more of the various types of environment variables maintained by a particular operating system. According to another alternative embodiment, an application embodying the present method receives a maximum thread indicator as an argument (step 60) included in the application launch argument list received by the application when it is launched.

Clean version:

[0019] Fig. 4 is a flow diagram that depicts additional alternative embodiments of a method for determining a maximum quantity of parallel processing threads. According to one such additional alternative embodiment, one or more environment variables (step 55) maintained by the operating system are utilized to establish a maximum quantity of parallel processing threads that an application embodying the present method is allowed to create. It is difficult to enumerate all of the various types of environment variables that can be maintained by any particular embodiment of an operating system. However, the scope of the claims appended hereto is intended to encompass derivative methods wherein the maximum quantity of parallel processing threads that can be created by an application is established according to one or more of the various types of environment variables maintained by a

particular operating system. According to another alternative embodiment, an application embodying the present method receives a maximum thread indicator as an argument (step 60) included in the application launch argument list received by the application when it is launched.